What Is Claimed Is:

1. A processor, comprising:

an instruction pipeline having N stages;

an instruction set, comprising a branch instruction and a branch notification instruction operative to receive at least one argument M; and

a loading module to place instructions in said instruction pipeline,

wherein said branch notification instruction is to indicate to said loading module via said at least one argument M that a branch instruction will occur within M instructions in said instruction pipeline, and wherein when said branch notification instruction is executed, said loading module is to load an instruction beginning at a branch point for said branch.

2. The processor of claim 1, wherein said branch notification instruction has at least two arguments comprising a number of instructions M, and a branch target.

3. The processor of claim 1, wherein said branch notification instruction is at least one of 16 bits or 32 bits.

4. A method of static branch prediction, comprising:

inserting a branch notification instruction into an instruction sequence before a branch instruction, wherein said branch notification instruction indicates a separation of M instructions from said branch instruction, and wherein said branch has a branch target;

executing said branch notification instruction; and

fetching an instruction starting at said branch target immediately after executing said branch notification instruction.

5. The method of claim 4, wherein said inserting a branch notification instruction further comprises: inserting a branch notification instruction indicating both a separation of M instructions from said branch instruction and a branch target.

6. The method of claim 4, wherein said inserting a branch notification instruction further comprises inserting one of a 16 bit and 32 bit branch notification instruction according to said separation M.

7. A machine-accessible medium containing software code that, when read by a computer, causes the computer to perform a method comprising:

inserting a branch notification processor instruction in an instruction sequence before a branch instruction, wherein said branch notification instruction indicates a separation of M instructions from said branch instruction, and wherein said branch instruction has a branch target;

executing said branch notification instruction; and

fetching an instruction starting at said branch target immediately after executing said branch notification instruction.

8. The machine-accessible medium of claim 7, wherein said inserting a branch notification instruction further comprises: inserting a branch notification instruction indicating both a separation of M instructions from said branch instruction and a branch target.

9. The machine-accessible medium of claim 7, wherein said inserting a branch notification instruction further comprises inserting one of a 16 bit and 32 bit branch notification instruction according to said separation M.

10. A system, comprising:

a random-access memory;

a processor coupled to said memory, said processor comprising:

an instruction pipeline having N stages;

an instruction set, comprising a branch instruction and a branch notification instruction operative to receive at least one argument M; and

a loading module to place said sequence of instructions in said instruction pipeline;

wherein said branch notification instruction is to indicate to said loading module via said at least one argument M that a branch instruction will occur within M instructions in said instruction sequence, and wherein when said branch notification instruction is executed, said loading module is to load an instruction beginning at a branch point for said branch.

11. The system of claim 10, wherein said branch notification instruction has at least two arguments comprising:

a number, which indicates the location of forthcoming branch instruction in said sequence of instructions; and

a branch target location.

12. The system of claim 10, wherein said branch notification instruction is at least one of 16 bits or 32 bits, depending on the offset needed to encode the branch target.